

Aufgabe 1

Beweisen Sie die Formel

$$\{true\} \quad x := 7; y := x + 3 \quad \{y = 10\}$$

im Hoare-Kalkül.

Idee von Tob:

kurze Regelkunde:

$$\{P[x \leftarrow e]\} \quad x := e \quad \{P\} \quad \text{(Zuweisung)}$$

Mit anderen Worten: Man ersetzt in der Nachbedingung alle vorkommen von x durch e .
 and here we go: (von hinten nach vorne, also von unten nach oben lesen ;))

$\{true\}$	$\{true\}$	
$\{true\}$	$\{7 + 3 = 10\}$	(Zuweisung)
$\{true\} \quad x := 7$	$\{x + 3 = 10\}$	(Zuweisung)
$\{true\} \quad x := 7; y := x + 3$	$\{y = 10\}$	

Aufgabe 2

Schreiben Sie ein WHILE'-Programm zur Berechnung der Signum-Funktion und beweisen Sie seine Korrektheit im Hoare-Kalkül.

Idee von Tob:

Erstmal die Regeln

$$\frac{\{I \wedge B\} \quad S \quad \{I\}}{\{I\} \quad \underline{while} \ B \ \underline{do} \ S \quad \{I \wedge \neg B\}} \quad \text{(while)}$$

```

1 sum:=0;
2 while not eof do
3   read x;
4   sum := sum + x;
5 output sum
    
```

Die Vorbedingung P ist relativ egal. Viel wichtiger ist die Nachbedingung Q , die muss den Zustand $in = \epsilon$ enthalten, da das Programm terminieren muss und der Wahrheitswert von eof davon abhängt. Im folgenden werden die Variablen *out* für die Ausgabe und *in* für die Eingabe verwendet. Das Symbol ϵ stellt hier eine leere Menge dar.

Die Invariante für die Whileschleife sollte eventuell nur Festhalten, dass die Ausgabe unverändert bleibt, die Schleifenbedingung lässt sich wie folgt definieren:

$$B = \{\neg(in = \epsilon)\} = \{in \neq \epsilon\} = \{in = [n_0, \dots, n_i]\}$$

$$\begin{aligned}
 P &= \{True\} \\
 \{ \} \quad sum &:= 0; & \{out = \epsilon \wedge in \neq \epsilon \wedge 0 + (n_0) \neq \epsilon\} \\
 \{ \} \quad \underline{while} \ \neg \underline{eof} \ \underline{do} & & \{out = \epsilon \wedge in \neq \epsilon \wedge sum + (n_0) \neq \epsilon\} \\
 \{ \} \quad \underline{read} \ x; & & \{out = \epsilon \wedge in = [n_1, \dots, n_i] \wedge sum + (n_0) \neq \epsilon\} \\
 \{ \} \quad sum &:= sum + x; & \{out = \epsilon \wedge in \neq \epsilon \wedge sum + x \neq \epsilon\} \\
 \{ \} \quad \underline{output} \ sum & & \{out = \epsilon \wedge in = \epsilon \wedge sum \neq \epsilon\} \\
 Q &= \{out = sum \wedge in = \epsilon\}
 \end{aligned}$$

Aufgabe 3

Führen Sie einen Korrektheitsbeweis unter Verwendung der axiomatischen Semantik zu folgendem Programm:

```
1 sum:=0;  
2 while not eof do  
3   read x;  
4   sum := sum + x;  
5 output sum
```

Aufgabe 4

Beweisen Sie die Gültigkeit des Axioms (A.4), d.h. zeigen Sie die Gültigkeit der Formel:

$$\{Q[output.T/output]\} \quad outputT \quad \{Q\}$$